



This is not a |

> #!/bin/bash

Redirecciones

Todos los comandos producen algún mensaje, ya sea de error o el resultado de la operación que realiza. Estos mensajes pasan a través de un flujo (stream) de datos llamado **stderr** (de standard error) y **stdout** (de standard output), el primero para el error y el segundo para el resultado del comando.

Adicionalmente la información que es requerida por el comando se toma del flujo llamado **stdin** (de standard input).

Las redirecciones redefinen a donde van los flujos de salida (stdout y stderr) y de donde proviene el de entrada.

```
$ du -csh ~/* > home.log
```

```
$ cat home.log
```

```
$ asd > error.log
```

```
$ asd 2> error.log
```

```
$ cat error.log
```

pipes:

```
$ ls /home/ | wc -l
```

> **#!/bin/bash**

Expansiones

`~` = `/home/user`

`*` = `all`

Expansión	Descripción	Ejemplo
<code>\$((expresion))</code>	Evalúa una expresión aritmética	<code>\$ echo \$((2**8))</code>
<code>{a,b,c}</code>	Crea una colección de argumetos con a, b y c	<code>\$ mkdir folder_{a,b,c}</code>
<code>{1..5}</code>	Crea una colección de argumentos con los números del 1 al 5	<code>\$ touch photo_{1..5}</code>
<code>{M..D}</code>	También funciona con letras	<code>\$ echo {M..D}</code>

> #!/bin/bash

Expansiones

~ = /home/user

* = all

Expansión

```
$((expresion))
```

Descripción

Evalúa una expresión aritmética

Ejemplo

```
$ echo $(( 2**8 ))
```

```
$ echo $(( 2/3 ))
```

```
$ echo "2/3" | bc
```

```
$ echo "scale=10;2/3" | bc
```

> **#!/bin/bash**

Expansiones

`~` = `/home/user`

`*` = `all`

Expansión	Descripción	Ejemplo
<code>\$((expresion))</code>	Evalúa una expresión aritmética	<code>\$ echo \$((2**8))</code>
<code>{a,b,c}</code>	Crea una colección de argumetos con a, b y c	<code>\$ mkdir folder_{a,b,c}</code>
<code>{1..5}</code>	Crea una colección de argumentos con los números del 1 al 5	<code>\$ touch photo_{1..5}</code>
<code>{M..D}</code>	También funciona con letras	<code>\$ echo {M..D}</code>

> **#!/bin/bash**

Encomillado

```
$ mkdir nombre apellido
```

```
$ echo hola mundo
```

```
$ echo 10$USD son 34mil$COP
```

```
$ mkdir "nombre apellido"
```

```
$ ls -l "nombre apellido"
```

```
$ echo "hola mundo"
```

```
$ echo "10$USD son 34mil$COP"
```

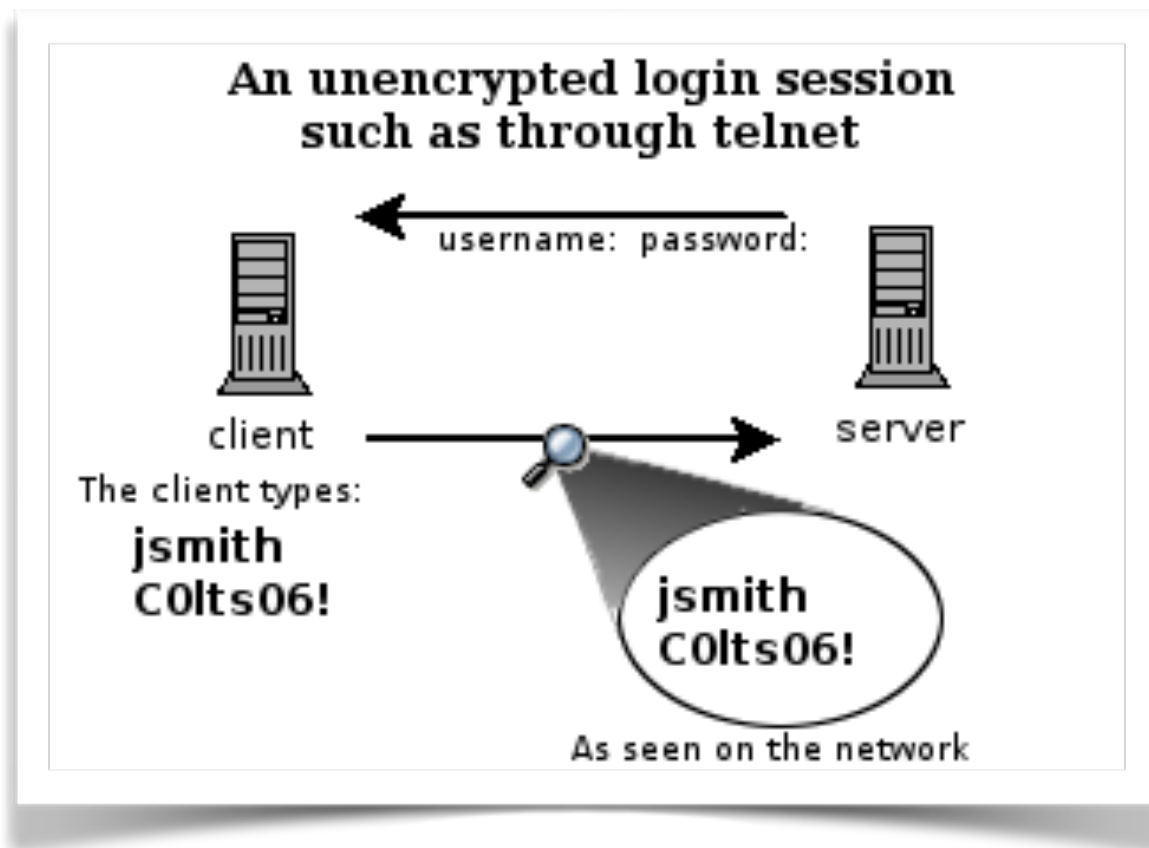
\$ \ ` son caracteres especiales, se usan comillas simples para retirar su función

```
$ echo '10$USD son 34mil$COP'
```

```
$ echo "10$USD son 34mil\$COP"
```

> **#!/bin/bash**

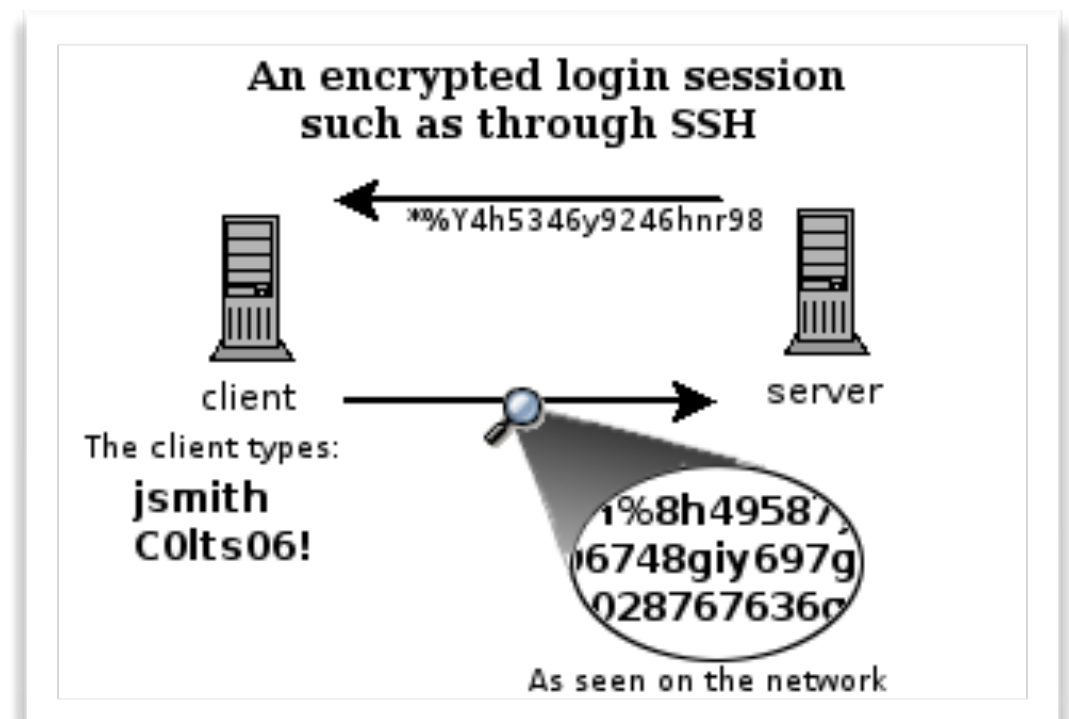
Conexiones remotas



telnet: TCP/IP

```
$telnet towel.blinkenlights.nl
```

ssh: security shell openSSH



> #!/bin/bash

Conexiones remotas con SSH

Copie la ip de su compañero (eth0):

```
$/sbin/ifconfig
```

Accediendo al shell remoto

```
$ssh username@server
```

Agregue al startup file el alias

```
alias ls="echo 'has sido  
hackeado';sl -a; ls"
```

```
$exit
```

Localmente examine el del startup file y borre la intrusión

